Data Quality Analysis of AIS data

International Association of Marine Aids to Navigation and Lighthouse Authorities

Kiki Beumer

Supervisor: Omar Eriksson

Contents

1 Introduction

The Automatic Identification System (AIS) data is vital for maritime navigation and safety. In a previous report, referenced here, we decoded the AIS data. In this report, we will explore the various data types, conducting a general analysis to better understand the dataset. Specifically, we will assess the quality and coverage of AIS data within a defined region. This assessment includes evaluating the geographical coverage, identifying error messages, examining position accuracy, and analyzing temporal distributions. We will also assess the time intervals between consecutive messages sent by vessels. Additionally, deviations in vessel routes will be visualized using GeoPandas maps to identify any anomalies. The dataset for this analysis was collected from the Papua New Guinea region between October 28, 2023, and January 26, 2024.

2 Objective

The quantitative analysis of AIS data involves examining the frequency, distribution, and statistical properties of various AIS message types within a specified region. To ensure reproducibility, tools for statistical analysis have been developed. The Python code can be found with the Github link in the references.

The first chapter focuses on assessing data quality by analyzing distributions, trends, and errors in the AIS messages. This includes checking for inaccuracies such as vessels incorrectly reported as being on land and evaluating positional accuracy. In the second chapter, the focus shifts to the geographic coverage of AIS data. The aim is to identify dead zones within the region by comparing actual message frequencies with those predicted based on vessel speed over ground. Extended intervals between messages may indicate areas with poor AIS coverage.

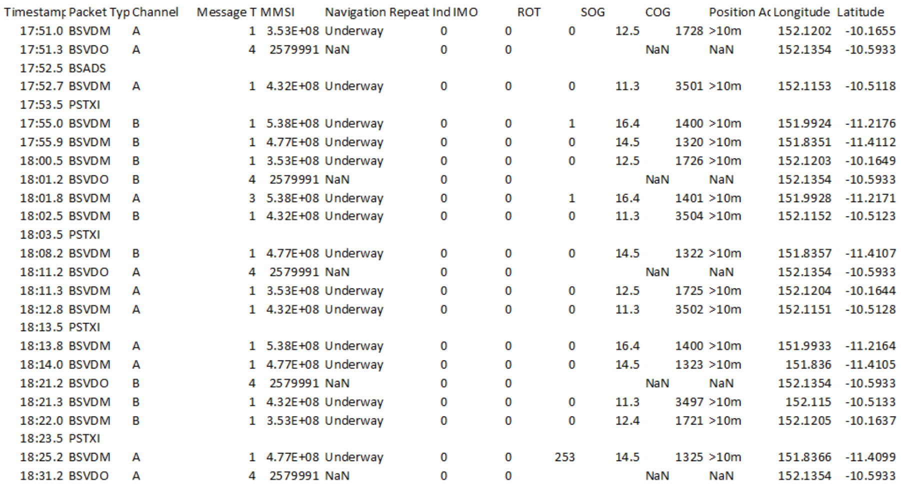
The final chapter compares the time intervals requirements between consecutive messages and that of the dataset. This includes predicting speed over ground from location data (longitude and latitude). These predictions will be assessed against actual data to ensure they meet regulatory requirements and accurately reflect vessel behavior. This analysis will identify deviations and serve as a tool for evaluating the quality of the AIS data.

3 Data Types

The subset used for this specific analysis is of size 6.599.735. This data was obtained from 28 October 2023 and 26 January 2024. The AIS messages have been decoded with the code mentioned in the Github link in the references. The csv file looks as follows:

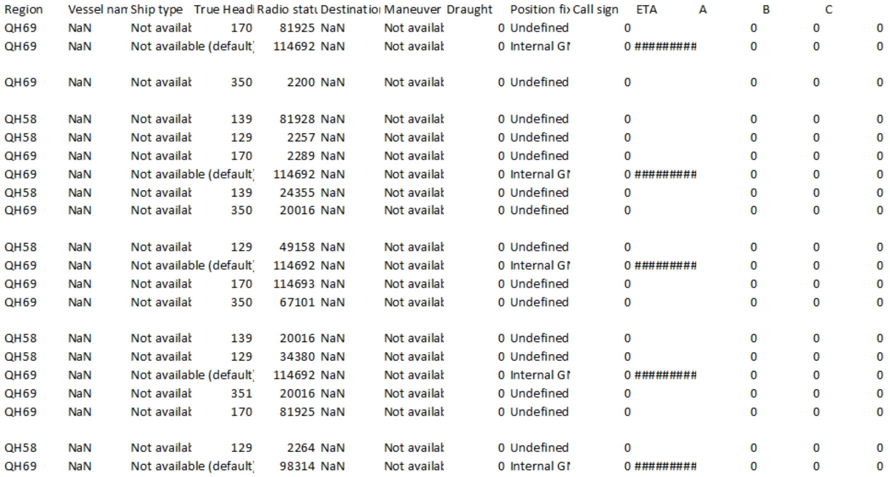
The following data types have been extracted from the AIS messages:

The distribution of the following quantitative data types was studied:
- ROT (rate of turn)



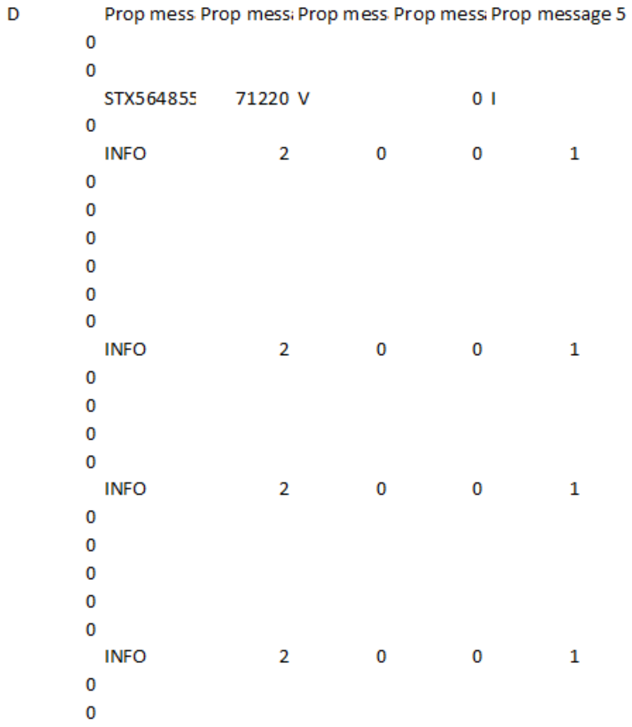
Timestamp	Packet Typ	Channel	Message T	MMSI	Navigation	Repeat Ind	IMO	ROT	SOG	COG	Position Ac	Longitude	Latitude
17:51.0	BSVDM	A	1	3.53E+08	Underway	0	0	0	12.5	1728	>10m	152.1202	-10.1655
17:51.3	BSVDO	A	4	2579991	NaN	0	0			NaN	NaN	152.1354	-10.5933
17:52.5	BSADS												
17:52.7	BSVDM	A	1	4.32E+08	Underway	0	0	0	11.3	3501	>10m	152.1153	-10.5118
17:53.5	PSTXI												
17:55.0	BSVDM	B	1	5.38E+08	Underway	0	0	1	16.4	1400	>10m	151.9924	-11.2176
17:55.9	BSVDM	B	1	4.77E+08	Underway	0	0	0	14.5	1320	>10m	151.8351	-11.4112
18:00.5	BSVDM	B	1	3.53E+08	Underway	0	0	0	12.5	1726	>10m	152.1203	-10.1649
18:01.2	BSVDO	B	4	2579991	NaN	0	0			NaN	NaN	152.1354	-10.5933
18:01.8	BSVDM	A	3	5.38E+08	Underway	0	0	1	16.4	1401	>10m	151.9928	-11.2171
18:02.5	BSVDM	B	1	4.32E+08	Underway	0	0	0	11.3	3504	>10m	152.1152	-10.5123
18:03.5	PSTXI												
18:08.2	BSVDM	B	1	4.77E+08	Underway	0	0	0	14.5	1322	>10m	151.8357	-11.4107
18:11.2	BSVDO	A	4	2579991	NaN	0	0			NaN	NaN	152.1354	-10.5933
18:11.3	BSVDM	A	1	3.53E+08	Underway	0	0	0	12.5	1725	>10m	152.1204	-10.1644
18:12.8	BSVDM	A	1	4.32E+08	Underway	0	0	0	11.3	3502	>10m	152.1151	-10.5128
18:13.5	PSTXI												
18:13.8	BSVDM	A	1	5.38E+08	Underway	0	0	0	16.4	1400	>10m	151.9933	-11.2164
18:14.0	BSVDM	A	1	4.77E+08	Underway	0	0	0	14.5	1323	>10m	151.836	-11.4105
18:21.2	BSVDO	B	4	2579991	NaN	0	0			NaN	NaN	152.1354	-10.5933
18:21.3	BSVDM	B	1	4.32E+08	Underway	0	0	0	11.3	3497	>10m	152.115	-10.5133
18:22.0	BSVDM	B	1	3.53E+08	Underway	0	0	0	12.4	1721	>10m	152.1205	-10.1637
18:23.5	PSTXI												
18:25.2	BSVDM	A	1	4.77E+08	Underway	0	0	253	14.5	1325	>10m	151.8366	-11.4099
18:31.2	BSVDO	A	4	2579991	NaN	0	0			NaN	NaN	152.1354	-10.5933

Figure 1: Output csv file



Region	Vessel nam	Ship type	True Headi	Radio statu	Destinatio	Maneuver	Draught	Position fi	Call sign	ETA	A	B	C
QH69	NaN	Not availak	170	81925	NaN	Not availak	0	Undefined	0		0	0	0
QH69	NaN	Not available (default		114692	NaN	Not availak	0	Internal GI	0	#########	0	0	0
QH69	NaN	Not availak	350	2200	NaN	Not availak	0	Undefined	0		0	0	0
QH58	NaN	Not availak	139	81928	NaN	Not availak	0	Undefined	0		0	0	0
QH58	NaN	Not availak	129	2257	NaN	Not availak	0	Undefined	0		0	0	0
QH69	NaN	Not availak	170	2289	NaN	Not availak	0	Undefined	0		0	0	0
QH69	NaN	Not available (default		114692	NaN	Not availak	0	Internal GI	0	#########	0	0	0
QH58	NaN	Not availak	139	24355	NaN	Not availak	0	Undefined	0		0	0	0
QH69	NaN	Not availak	350	20016	NaN	Not availak	0	Undefined	0		0	0	0
QH58	NaN	Not availak	129	49158	NaN	Not availak	0	Undefined	0		0	0	0
QH69	NaN	Not available (default		114692	NaN	Not availak	0	Internal GI	0	#########	0	0	0
QH69	NaN	Not availak	170	114693	NaN	Not availak	0	Undefined	0		0	0	0
QH69	NaN	Not availak	350	67101	NaN	Not availak	0	Undefined	0		0	0	0
QH58	NaN	Not availak	139	20016	NaN	Not availak	0	Undefined	0		0	0	0
QH58	NaN	Not availak	129	34380	NaN	Not availak	0	Undefined	0		0	0	0
QH69	NaN	Not available (default		114692	NaN	Not availak	0	Internal GI	0	#########	0	0	0
QH69	NaN	Not availak	351	20016	NaN	Not availak	0	Undefined	0		0	0	0
QH69	NaN	Not availak	170	81925	NaN	Not availak	0	Undefined	0		0	0	0
QH58	NaN	Not availak	129	2264	NaN	Not availak	0	Undefined	0		0	0	0
QH69	NaN	Not available (default		98314	NaN	Not availak	0	Internal GI	0	#########	0	0	0

Figure 2: Output csv file



D	Prop mess	Prop mess	Prop mess	Prop mess	Prop message 5
0					
0					
	STX56485	71220	V	0	I
0					
	INFO	2	0	0	1
0					
0					
0					
0					
0					
0					
	INFO	2	0	0	1
0					
0					
0					
0					
	INFO	2	0	0	1
0					
0					
0					
0					
0					
	INFO	2	0	0	1
0					
0					

Figure 3: Output csv file

Table 1: AIS Data Types

Columns in *.csv file	Description	Data Type	Example
Timestamp	Timestamp from the AIS basestation	datetime	31/12/2015 23:59:59
Packet Type	Defines the packet type, which can be defined in different formats	string	AIVDM
Channel	Radio channel code	string	A
Message Type	Specifies the kind of information being transmitted	float	1
MMSI	MMSI number of the vessel	string	538009733
Navigation Status	Navigational status from AIS message if available	string	Under way using engine
Repeat Indicator	Message repeat count	float	0
IMO	IMO number of the vessel	float	9675626
ROT	Rate of turn in degrees/min	string	2.1873
SOG	Speed over ground in knots	string	12.5
COG	Course over ground relative to true north	float	1322
Position Accuracy	Accuracy of the DGPS-quality fix	string	>10m
Longitude	Geographic coordinate	float	152.120208333333
Latitude	Geographic coordinate	float	-10.1654516666666
Region	Region based on IARU Locator grid system	string	QH
Vessel name	Name of vessel	string	AAL HONG KONG
Ship Type	Type of vessel	string	Cargo, all ships of this type
True Heading	Heading of the vessel in degrees	string	170
Radio status	State of the vessel's AIS transceiver	float	81925
Destination	Intended port of arrival or destination of the vessel	float	NaN
Maneuver Indicator	Vessel's current maneuvering status	string	Special maneuver
Draught	Vessel's vertical distance between the waterline and the bottom of the hull in meters	float	17.5
Position Fixed type	Type of positional fixing device from the AIS message	string	Internal GNSS
Call sign	unique identifier assigned to a vessel's radio communication equipment	float	3E2334
ETA	Estimated time of arrival	datetime	10/28/2023 7:12:00 AM
A	Length from GPS to the bow in meters	float	202
B	Length from GPS to the stern in meters	float	33
C	Length from GPS to starboard side in meters	float	13
D	Length from GPS to port side in meters	float	25
Error	Any error messages that appear	string	Checksum mismatch

- SOG (speed over ground)
- COG (course over ground)
- Draught
- Longitude
- Latitude
- A (length from GPS to the bow in meters)
- B (length from GPS to the stern in meters)
- C (length from GPS to starboard side in meters)
- D (length from GPS to port side in meters)

The information available in the "Message Type" column results in a significant number of 'NaN' values, which will be dropped to obtain a more informative subset.

4 Quantitative Analysis per Region

The following code can be used to reproduce the histogram and observe the distribution of the quantitative variables. The lineplot through the histogram is the KDE. This is a non-parametric way to estimate the probability density function (pdf) of a random variable.

```
# Exchange 'Draught' with any other column name if needed
ais_draught = ais.dropna(subset='Draught')
ais_draught = ais_draught[ais_draught['Draught'] != 0.0]

sns.set_theme(style="darkgrid")
sns.histplot(data=ais_draught, x="Draught", kde=True)
```

```
plt.show()
```

Listing 1: Code Histogram and KDE

4.1 Speed over Ground

Values where `SOG == 102.3` are dropped, since it is the numerical equivalent of 'NaN' for this variable. The resulting size of the sample is 4.597.463. KDE shows that this distribution is approximately normal.

To closely examine the speed over ground in a specific area, we can subset the data for a particular Maidenhead region; QH69. This region is more precisely defined by the two numbers following the two capital letters that identify the general area.

```
#Subset for a region
ais_sog = ais_sog[ais_sog["Region"]=='QH69']

#Create the histogram
sns.set_theme(style="darkgrid")
sns.histplot(data=ais_sog, x="SOG", kde=True)
plt.xlim(0, 25)
plt.show()

#Mean, SD, min ,max
print(ais_sog["SOG"].mean())
print(ais_sog["SOG"].std())
print(ais_sog["SOG"].min())
print(ais_sog["SOG"].max())

# OUTPUT:
# Mean: 12.386505500958968
# SD: 2.4601595797148024
# Min: 0.0
# Max: 99.6
```

Listing 2: Subsetting for region

Once we have subsetted the data, we can plot the SOG again. This detailed view can help identify regions where there might be other abnormal activity. Region QH69 shows no significant deviations from the original dataset. The mean and distribution appear consistent. The code to find additional statistics is included below as well.

4.2 Course over Ground

If `COG == 3600`, the numerical equivalent of 'NaN' for this variable, these values are dropped. The resulting size of the subset is 4.594.227. The histogram shows a trimodal distribution.

We can repeat the same subsetting method for region QH69 as we did with SOG. The resulting histogram is as follows:

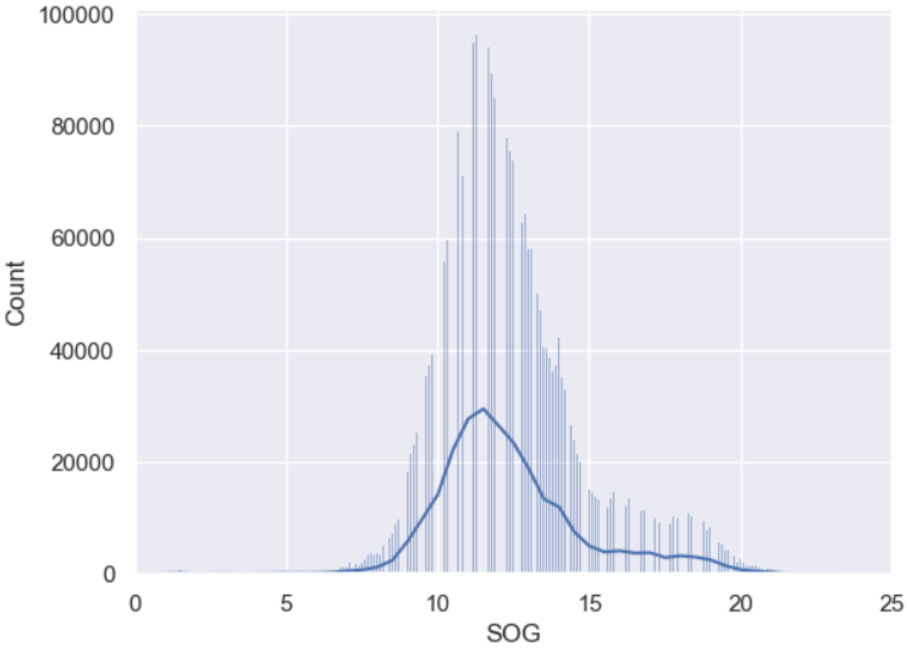


Figure 4:
Speed over ground distribution for region QH69

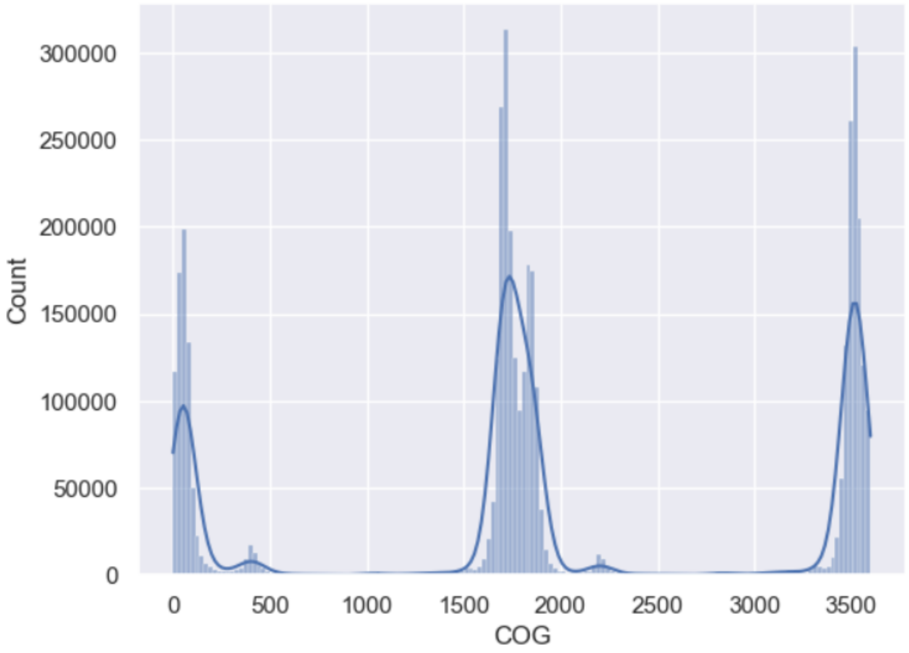


Figure 5:
Course over ground distribution for region QH69

Again, we see the same distribution as the complete dataset. This might yield more significantly different results if you have data points from a larger region compared to the ones we have here.

```
# OUTPUT:
# Mean: 1982.484944691728
# SD: 1215.848305429338
# Min: 0.0
# Max: 3599.0
```

Listing 3: Output COG in region QH69

4.3 Draught

The resulting size of the sample after dropping 'NaN' values is 117.568. Again , we follow the same steps.

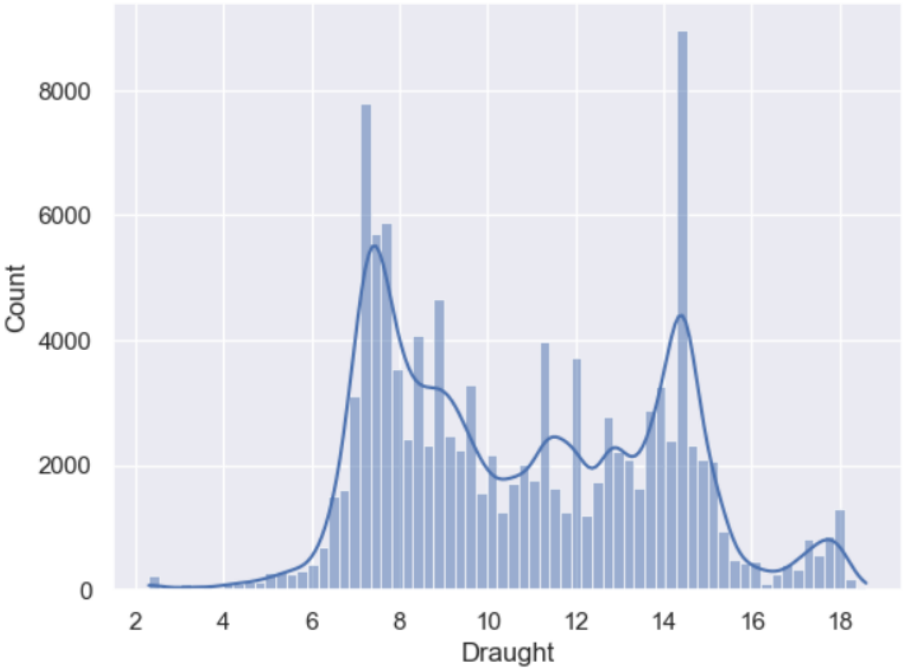


Figure 6:
Draught distribution

```
print(ais["Draught"].mean())
print(ais["Draught"].std())
print(ais["Draught"].min())
print(ais["Draught"].max())

# OUTPUT:
# Mean: 0.19248814289353253
# SD: 1.4908421152141382
# Min: 0.0
# Max: 18.6
```

Listing 4: Output COG in region QH69

4.4 Vessel's length

All rows where `A==0 | B==0 | C==0 | D==0` are dropped, this is the numerical equivalent of 'NaN' for this variable. The resulting size of the sample is 118.579. All length are displayed in one plot and additional statistics are included in table 2.

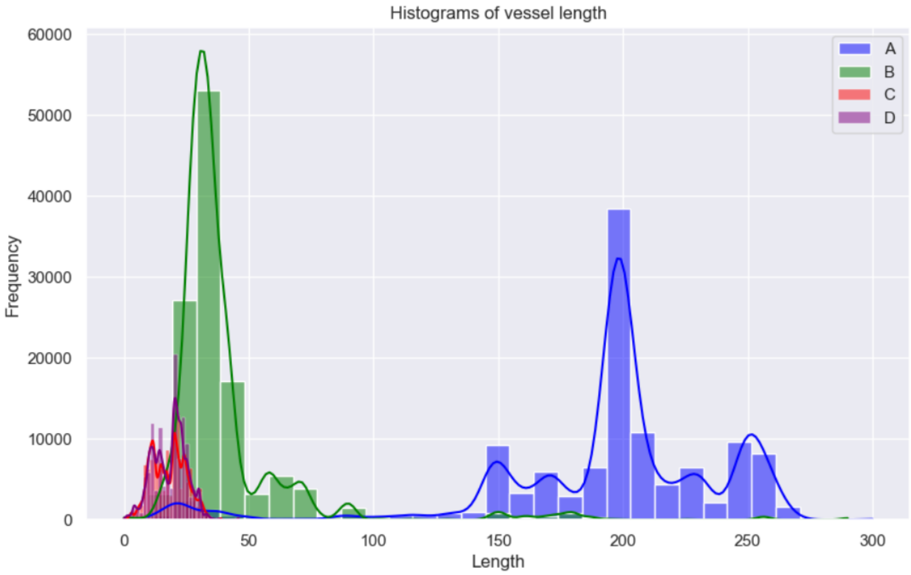


Figure 7:
Vessel's length distribution

Table 2: Statistics on vessel's length

	Length from GPS to the bow	Length from GPS to the stern	Length from GPS to starboard side	Length from GPS to port side
	Size A	Size B	Size C	Size D
Mean	193.77	40.15	18.0	18.42
SD	48.32	27.59	6.88	6.70
Min	9.0	1.0	1.0	1.0
Max	300.0	290.0	39.0	46.0

4.5 Unique vessels

Refers to the count of distinct vessels that are identifiable within the dataset based on their unique identifiers (Maritime Mobile Service Identity (MMSI) number). Each vessel is represented with a unique MMSI number, and counting the number of unique ships provides insight into the diversity and volume of maritime traffic captured by the dataset. With a simple `.nunique()` function we can get that this dataset is provided by 1973 unique vessels.

```
# Extract date from the timestamp
ais_cleaned[’date’] = ais_cleaned[’Timestamp’].dt.date

#Create table
unique_vessels = ais_cleaned.groupby(’date’)[’MMSI’].nunique().
    reset_index()
unique_vessels.columns = [’Date’, ’Unique vessels’]
```

Listing 5: Get Unique vessel code

To visualize the changes in traffic volume over time, we first create a pivot table or use a groupby operation. Then, plot this aggregated data. This allows us to observe that there is fluctuation in the number of vessels sending AIS messages each day/ month or year. There is no specific trend to be seen in this plot.

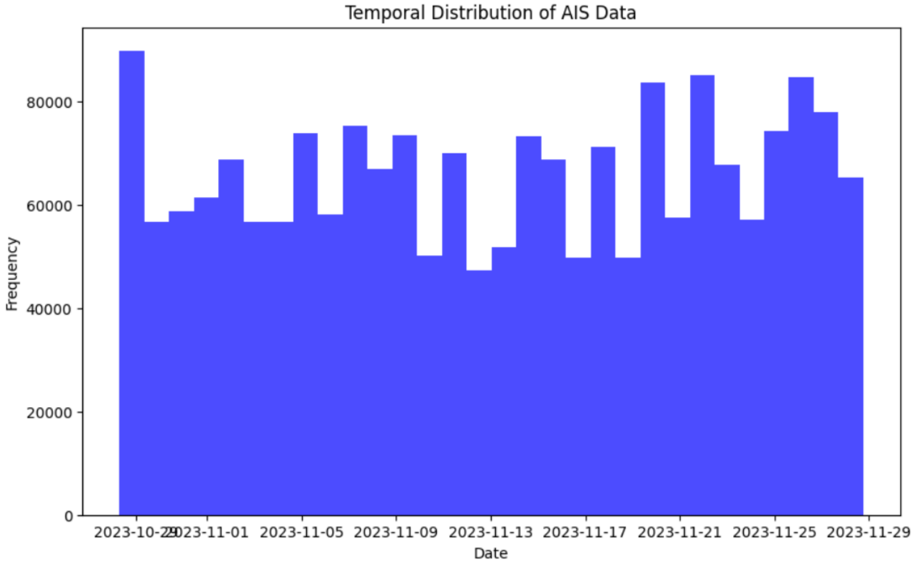


Figure 8: Unique vessel frequency over time

5 Geographical boundaries and coverage

5.1 Objective

This refers to the defined spatial limits within which the data was collected or is relevant. For AIS data, this involves specifying the latitude and longitude ranges that encompass the area covered by the dataset. This includes identifying the maximum and minimum longitude and latitude values, as well as checking for any missing values and identifying dead zones.

```
#Import libraries
from shapely.geometry import Point
import geopandas as gpd
from geopandas import GeoDataFrame
import geodatasets

# Set points x, y to the longitude and latitude values
geometry = [Point(xy) for xy in zip(ais[’Longitude’], ais[’Latitude’])]
gdf = GeoDataFrame(ais, geometry=geometry)

#Add the world map
gpd.read_file(gpd.datasets.get_path(’naturalearth_lowres’))
world = gpd.read_file(geodatasets.data.naturalearth.land[’url’])
gdf.plot(ax=world.plot(figsize=(12, 8)), marker=’o’, color=’red’,
    markersize=15)

world = gpd.read_file(geodatasets.data.naturalearth.land[’url’])

# Plot the world map
ax = world.plot(figsize=(10, 6))

# Plot the red
points on the map
gdf.plot(ax=ax, marker=’o’, color=’red’, markersize=15)

#Adjust the boundaries base don the region
ax.set_xlim(145, 155) #long
ax.set_ylim(-15, -5) #lat

# Show the plot
plt.show()
```

Listing 6: Plot the data points in geopandas map

This map would be more suitable if the dataset would contain points of more than one IARU Maidenhead region. However, it is helpful to visualize the location of the data points.

5.2 Limitations

To observe the geographical boundaries we can display the longitude and latitude in a scatter plot. The dotted red lines are the minimum and maximum longitude and latitude in the dataset. We can use a scatterplot to visualize the coverage of longitude and latitude within these boundaries.

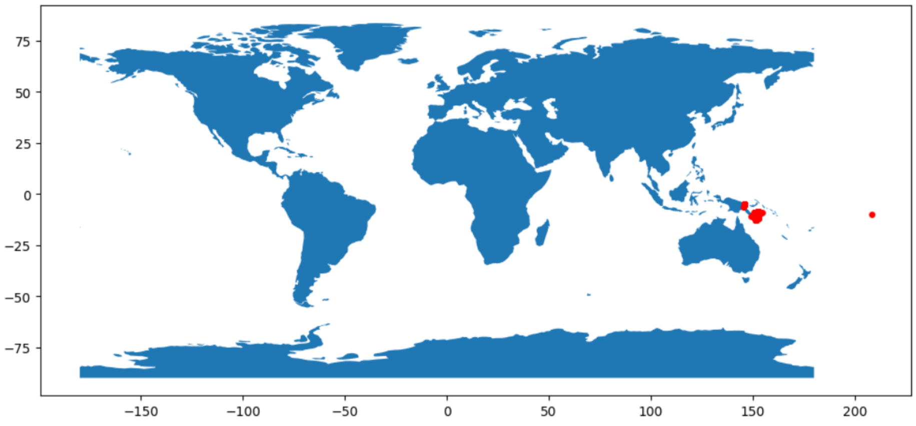


Figure 9:
Geopandas map of datapoints

However, this tool is not effective for identifying missing values, as the plot also includes longitude and latitude points that fall on land.

6 Data Quality in Region

Refers to analyzing the frequency or detail level of position reports for individual trips made by ships. This involves assessing how often position updates are recorded for each voyage undertaken by a ship. This includes the previously performed analysis as well, such as the geographical boundary and the temporal distribution analysis. To begin, we assess the dataset for errors by identifying entries with a ”Checksum mismatch.” Additionally, we evaluate the percentage of data points that have a position accuracy exceeding 10 meters.

6.1 Error checking

The provided code was designed to count the number of error messages within the dataset. However, no error messages were detected in this dataset. This outcome is attributed to the fact that proprietary AIS messages do not generate explicit error messages but instead utilize the ”Datafield” columns (shown in figure 12). Additionally, our analysis did not reveal any mismatched checksums or unexpected values in the data during decoding.

```
ais[ais["Error"].notna()]["Error"].value_counts()

#OUTPUT:
# 0
```

Listing 7: Python code for main

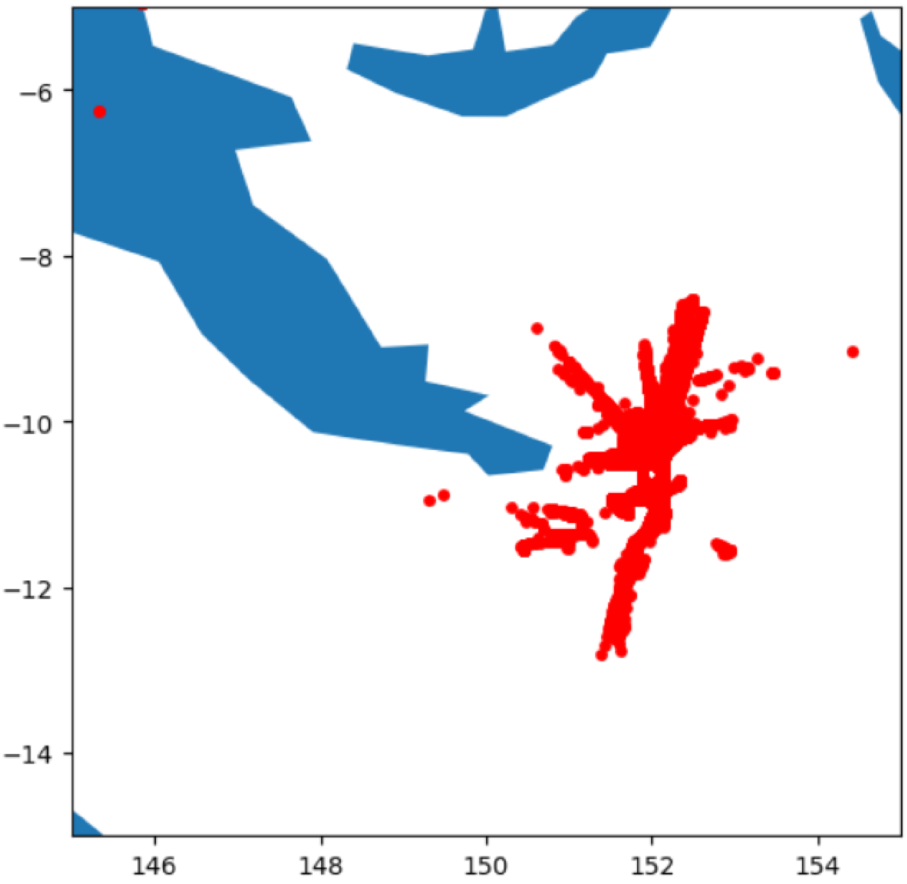


Figure 10:
Geopandas map of datapoints

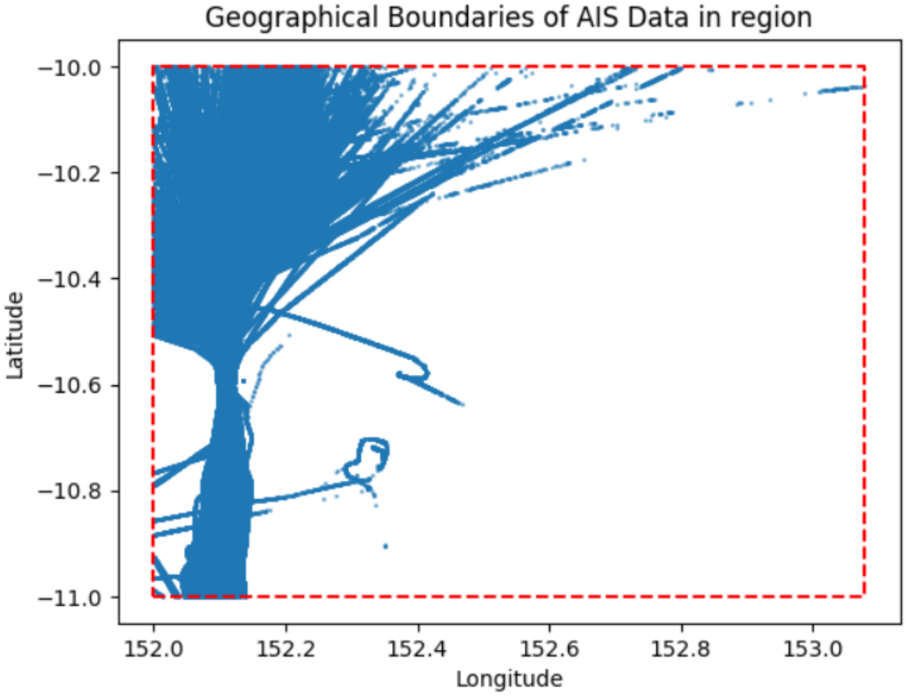
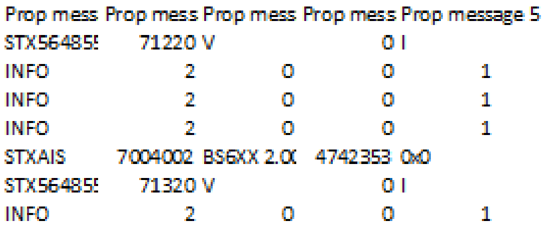


Figure 11: Geographical coverage of dataset



Prop mess	Prop mess	Prop mess	Prop mess	Prop message 5
STX564855	71220	V	0	I
INFO	2	0	0	1
INFO	2	0	0	1
INFO	2	0	0	1
STXAIS	7004002	BS6XX 2.0(	4742353	0x0
STX564855	71320	V	0	I
INFO	2	0	0	1

Figure 12: Added columns in dataset for proprietary messages

6.2 Position Accuracy

When we count the values in the ”Accuracy Position” column, we find that 65.14% of the total data has a position accuracy greater than 10 meters. It's important to note that position accuracy is not always provided, depending on the message type. Among the available position accuracy values, 93.48% of the data has a position accuracy greater than 10 meters.

We can subset the data for just the values that had a position accuracy of >10m. Then we can plot these points into the geopandas map to see the corresponding routes.

Table 3: Position Accuracy per MMSI count table

Position Accuracy	**0**	**<10m**	**>10m**
MMSI			
0	163849	0	0
125962272	0	0	38
209034000	0	0	1963
209669000	0	0	1994
209691000	0	0	43
...	...	...	...
636093180	0	1916	0
636093157	0	0	34
6360931800	0	130	1732

7,54% vessels of the total messages were able to transmit AIS messages with a position accuracy smaller than 10 meters. Which off course means that 92,46% of the vessels only transmitted messages with their position accuracy exceeding 10 meters.

To visualize data points with values either below or above 10 meters, you can use the following code.

```
#Subset the data for position accuracies larger than 10 meters
ais_pos_acc = ais_subset[ais_subset["Position Accuracy"]==">10m"]

#Import libraries
from shapely.geometry import Point
import geopandas as gpd
from geopandas import GeoDataFrame
import geodatasets
import matplotlib.pyplot as plt

#Set points x,y to the longitude and latitude values
geometry = [Point(xy) for xy in zip(ais_pos_acc[’Longitude’],
    ais_pos_acc[’Latitude’])]
gdf = GeoDataFrame(ais_pos_acc, geometry=geometry)

# Get world map
world = gpd.read_file(geodatasets.data.naturalearth.land[’url’])
```

```
gdf.plot(ax=world.plot(figsize=(12, 8)), marker=’o’, color=’red’,
    markersize=15)

# Plot the world map
ax = world.plot(figsize=(10, 6))

# Plot the red points on the map
gdf.plot(ax=ax, marker=’o’, color=’red’, markersize=15)

#Adjust the boundaries of the map based on the region
ax.set_xlim(145, 155) #long
ax.set_ylim(-15, -5) #lat

#Show the plot
plt.show()
```

Listing 8: Code for geopandas map for subsetted data

6.3 On land values

The global land mask library can calculate whether the vessel was on land or not based on the longitude and latitude provided. This accuracy is between 100 meter and 1 kilometer. The Global Land Mask library is suitable for general applications. It may not be perfect for applications requiring very fine spatial accuracy near coastlines or in areas with complex geographic features such as inland waters.

```
# pip install global_land_mask (if needed)
from global_land_mask import globe

# Apply the globe.is_land function to create the ’Is Land’ column
ais[’Is Land’] = ais.apply(lambda row: globe.is_land(row[’Latitude’],
    row[’Longitude’]), axis=1)

#Show results
ais.head()
```

Listing 9: Code to find position reports on land

Creating a count table of these values reveals that, in this subset of the dataframe, there are no instances where a vessel was recorded as being on land. While this observation might be useful for identifying potential errors, it cannot be conclusively interpreted, as there could be other explanations for this result.

```
#Create count table of "Is Land" values
island_table = ais[’Is Land’].value_counts().reset_index()
island_table.columns = [’Is Land’, ’Count’]

#Show the results
island_table
```

Listing 10: Code to create table for ”Is Land” values

This code produces a new column in the dataset *ais*; ”Is Land”. This column gives the boolean value ”True” is the longitude and latitude point were on

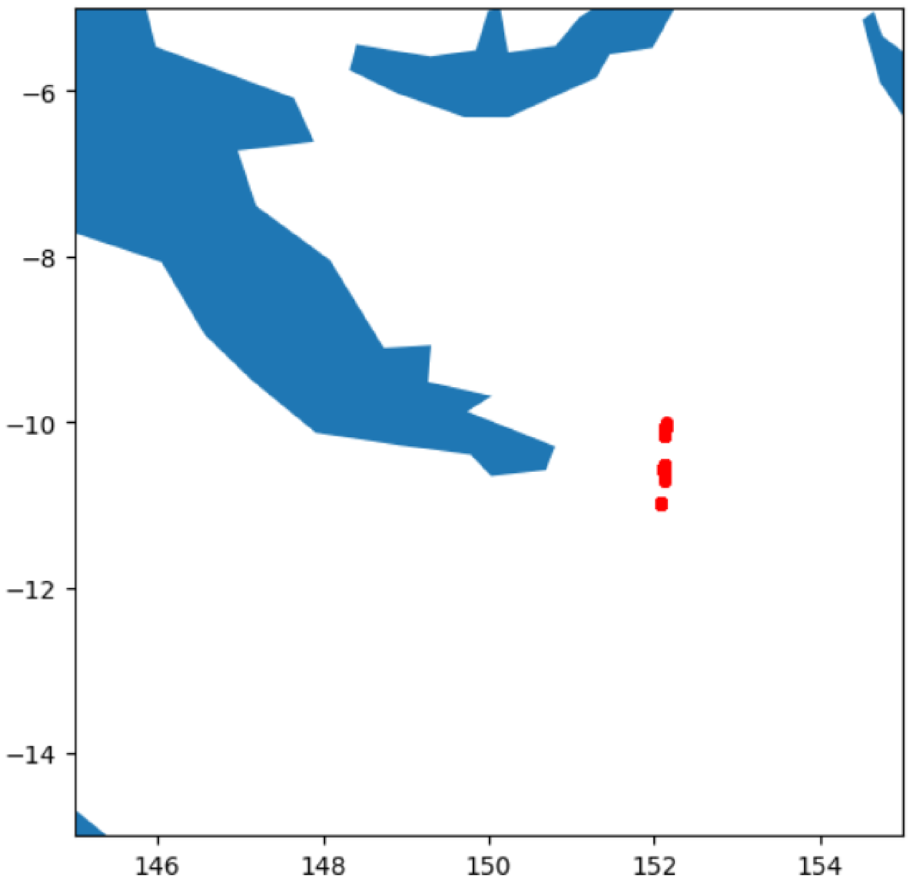


Figure 13: Geopandas map for subsetted data; position accuracy >10m

land. Similarly the value ”False”, indicates that this vessel was on water when transmitting the message.

Table 4: AIS Data Types

Is Land	**Count**
False	118.899
True	0

7 Message frequency based on SOG and ROT

First, we will look at the distribution of the time frequency between consecutive AIS messages. Created this extra column is easily done with a `.diff` function between consecutive rows, since the data is already sorted by timestamp. However we do need to groupby based on the MMSI number to ensure that the consecutive messages were send by the same vessel. The `pd.infer_freq` function shows that the time between consecutive messages of a vessel is not constant. Then plotting this data shows that the time interval has a right skewed distribution.

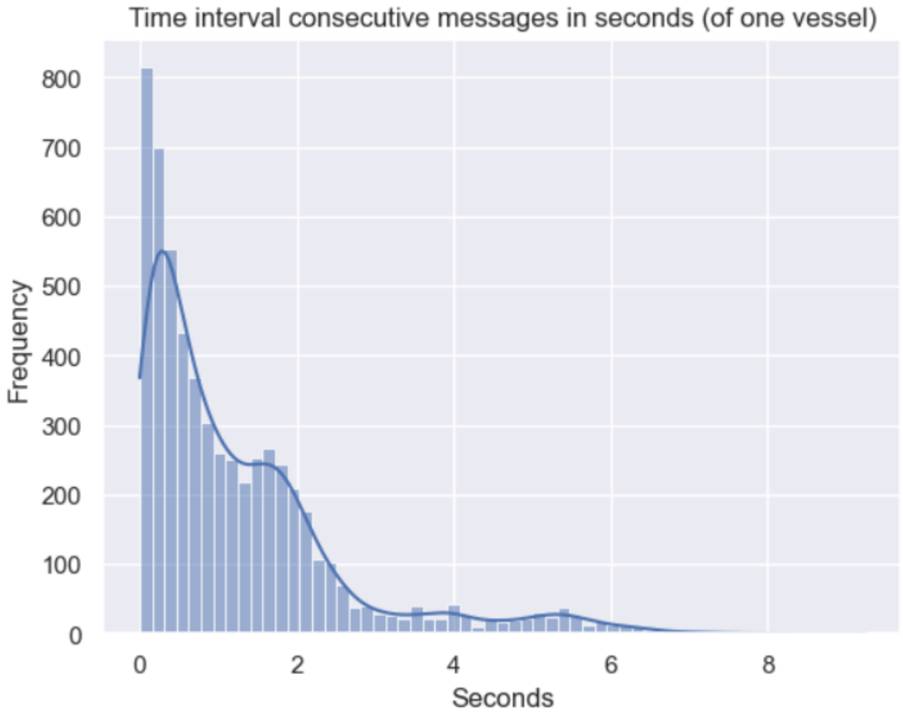
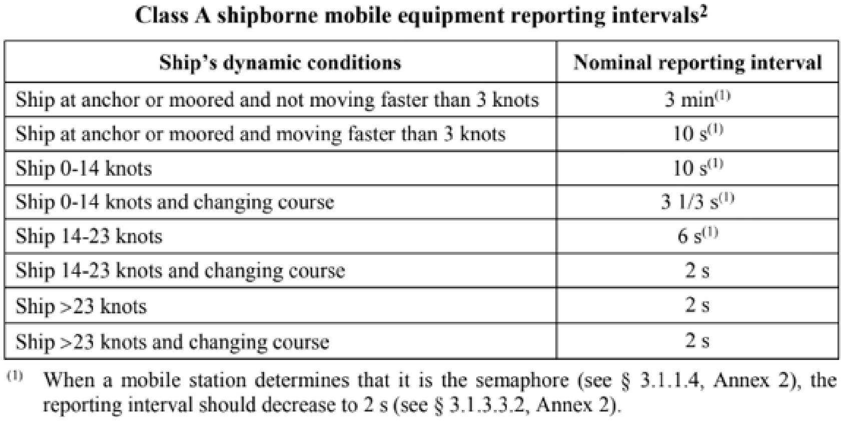


Figure 14: Time interval frequency

According to the requirements outlined in Figure 14, a vessel must transmit AIS messages at specified intervals based on its Speed Over Ground (SOG) and Rate of Turn (ROT). To address this, first subset the data to categorize the different groups and determine the required transmission frequency for each. Then, plot the resulting subsets on a map to visually identify locations where the transmission requirements were not met. Subsetting can be adjusted to accommodate different rules that need to be checked.



Class A shipborne mobile equipment reporting intervals[2]

Ship's dynamic conditions	Nominal reporting interval
Ship at anchor or moored and not moving faster than 3 knots	3 min(1)
Ship at anchor or moored and moving faster than 3 knots	10 s(1)
Ship 0-14 knots	10 s(1)
Ship 0-14 knots and changing course	3 1/3 s(1)
Ship 14-23 knots	6 s(1)
Ship 14-23 knots and changing course	2 s
Ship >23 knots	2 s
Ship >23 knots and changing course	2 s

(1) When a mobile station determines that it is the semaphore (see § 3.1.1.4, Annex 2), the reporting interval should decrease to 2 s (see § 3.1.3.3.2, Annex 2).

Figure 15: Time interval frequency requirements

```
# Ensure that the "SOG" column is numerical type for the subsetting
ais["SOG"] = pd.to_numeric(ais["SOG"], errors=’coerce’)

# Ship at anchor or moored and not moving faster than 3 knots
ais_anchor_less_3knots = ais[
    (ais["SOG"] < 3.0) &
    ((ais["Navigation Status"] == ’At anchor’) |
     (ais["Navigation Status"] == ’Moored’))
]

# Ship at anchor or moored and moving faster than 3 knots
ais_anchor_more_3knots = ais[
    (ais["SOG"] > 3.0) &
    ((ais["Navigation Status"] == ’At anchor’) |
     (ais["Navigation Status"] == ’Moored’))
]

# Ship 0-14 knots
ais_0_14 = ais[
    ((ais["SOG"] > 0.0) &
     (ais["SOG"]<= 14.0))
     &
    ((ais["Navigation Status"] != ’At anchor’) |
     (ais["Navigation Status"] != ’Moored’))
     &
    (ais["ROT"] == 0)
]

ais_0_14_turning = ais[
    ((ais["SOG"] > 0.0) &
     (ais["SOG"]<= 14.0))
     &
    ((ais["Navigation Status"] != ’At anchor’) |
     (ais["Navigation Status"] != ’Moored’))
     &
    (ais["ROT"] != 0)
]
ais_14_23 = ais[
    ((ais["SOG"] > 14.0) &
     (ais["SOG"]<= 23.0))
     &
    ((ais["Navigation Status"] != ’At anchor’) |
     (ais["Navigation Status"] != ’Moored’))
     &
    (ais["ROT"] == 0)
]

ais_14_23_turning = ais[
    ((ais["SOG"] > 14.0) &
     (ais["SOG"]<= 23.0))
     &
    ((ais["Navigation Status"] != ’At anchor’) |
     (ais["Navigation Status"] != ’Moored’))
     &
    (ais["ROT"] != 0)
]

```

```
ais_more_23 = ais[
    (ais["SOG"] > 23.0)
     &
    ((ais["Navigation Status"] != 'At anchor') |
     (ais["Navigation Status"] != 'Moored'))
     &
    (ais["ROT"] == 0)
]

ais_more_23_turning = ais[
    (ais["SOG"] > 23.0)
     &
    ((ais["Navigation Status"] != 'At anchor') |
     (ais["Navigation Status"] != 'Moored'))
     &
    (ais["ROT"] != 0)
]
```

Listing 11: Subsetting data based on requirements

After subsetting the data we check the size of the resulting dataset and plot this data in the map.

```
#Subset for values that did not meet requirements
ais_anchor_less_3knots[ais_anchor_less_3knots["minutes"]>3.0]

# Import libraries
from shapely.geometry import Point
import geopandas as gpd
from geopandas import GeoDataFrame
import geodatasets
import matplotlib.pyplot as plt

# Follow same steps as previously
geometry = [Point(xy) for xy in zip(ais_anchor_less_3knots['Longitude'
    ], ais_anchor_less_3knots['Latitude'])]
gdf = GeoDataFrame(ais_anchor_less_3knots, geometry=geometry)

world = pd.read_file(geodatasets.data.naturalearth.land['url'])
gdf.plot(ax=world.plot(figsize=(12, 8)), marker='o', color='red',
    markersize=15)

world = gpd.read_file(geodatasets.data.naturalearth.land['url'])

# Plot the world map
ax = world.plot(figsize=(10, 6))

# Plot your points on the map
gdf.plot(ax=ax, marker='o', color='red', markersize=15)

# Adjust the limits depending on the region
ax.set_xlim(145, 155)
ax.set_ylim(-15, -5)

# Show the plot
plt.show()
```

Listing 12: Code to plot points that did not meet frequency requirements

Figure 16 illustrates an example of the data points that failed to meet the required frequency.

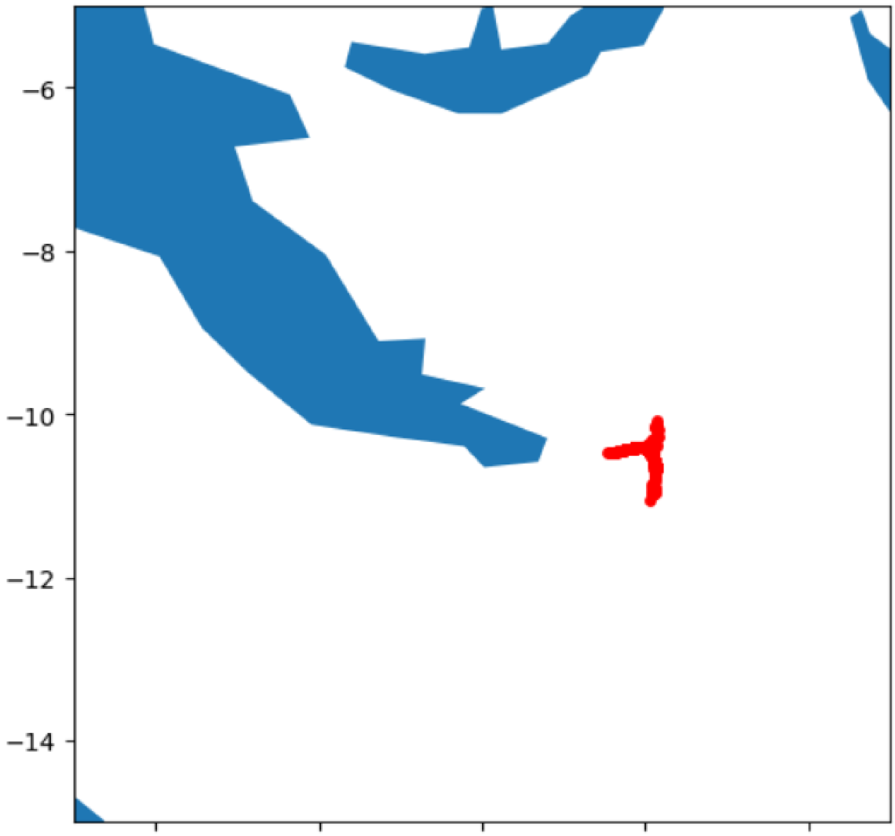


Figure 16:
Geopandas map of datapoints

8 Dead zones

The GitHub file contains the complete code necessary for categorizing data into three distinct zones: ”No Deadzone,” ”Possible Deadzone,” and ”Deadzone.” The code assigns colors to these categories for better visualization on a map. Specifically:

”No Deadzone” is assigned if the time interval between consecutive messages is less than 45 seconds. ”Possible Deadzone” applies to intervals of less than 60 seconds. ”Deadzone” is designated for intervals exceeding 1 minute. These thresholds are provided as examples, and can be adjusted based on specific research needs.

Figure 17 shows the result of the code. The x and y limits of the right plot can be adjusted based on the region to research.

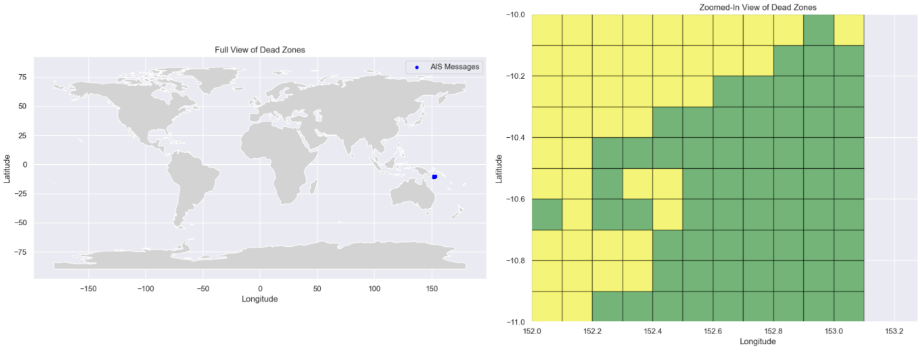


Figure 17:
Color coded Geopandas map for deadzone visualization

9 Limitations and Concluding Insights on AIS Data Quality

Ensuring that AIS data comprehensively covers all vessels within a defined geographical area is challenging. Smaller vessels may not be included in the AIS data due to system limitations. Analyzing dead zones—areas can help identify regions with potential gaps. However, this method alone does not fully capture the extent of inaccuracies in the data.

To assess the frequency of position reports, we compare the actual update intervals against the required intervals. Dead zones can be inferred from instances where the data update frequency fails to meet the required intervals.

Outlier detection can identify deviations in the AIS data. However, without global information for expected vessel speeds and other parameters, detecting and evaluating outliers relies solely on the available data.

Evaluating the spatial accuracy of AIS data involves comparing reported positions to known geographic features. While AIS systems often provide predefined accuracy levels based on various factors, these may not always align with actual positional accuracy. Quantifying spatial accuracy can be limited.

10 Possible future approaches

AIS data could provide a more comprehensive overview by filling coverage gaps and identifying regions mistakenly reported as dead zones. By incorporating more advanced and cost-effective AIS transponders, smaller vessels can also be more accurately represented in the data, ensuring that both large and small vessels are effectively tracked. This would lead to a more complete and reliable picture of maritime activity, improving safety and coverage across all vessel sizes and regions. By improving machine learning algorithms as well, the identification and analysis of dead zones will become more accurate and reliable.

By regulating and examining the position accuracy in AIS messages, we can verify and improve the accuracy of the reported data. This will enable machine learning algorithms to predict routes, identify dead zones, and detect deviating behavior, which will lead to better data quality assessment. Establishing global standards for outlier detection will reduce inconsistencies in AIS data interpretation as well.

Frequently updating regulations worldwide to align with technological developments, would ensure that all vessels follow the latest AIS standards. By addressing these limitations, AIS data quality and reliability can be improved, leading to safer and more efficient navigation.

11 Conclusion

Geographical boundaries are useful for exploring the dataset's spatial coverage, allowing for an evaluation of data quality in a region. Position accuracy is another valuable tool for assessing data quality, providing insights into the precision of reported locations.

Comparing the required message frequency based on parameters such as Speed Over Ground (SOG) and Rate of Turn (ROT) can also be a useful method for assessing data quality. This approach can be tailored to meet specific requirements and regional conditions.

Overall, quantifying AIS data quality involves tackling several challenges, including incomplete vessel reporting, data gaps in coverage areas, compliance with time intervals, outlier detection, and spatial accuracy. While each assessment method offers valuable insights, they also have their limitations. Continued research and the inclusion of additional data are important for achieving a more precise and thorough understanding of AIS data quality.

12 References

IALA. Retrieved from:
https://www.iala-aism.org/
AIS Sentence Decoding report. Retrieved from:
https://github.com/kikibeumer/Data-Quality-analysis-if-AIS-dataset/blob/main/AIS_sentence_decoding_Kiki_Beumer_IALA.pdf

AIS message decoder on Github. Retrieved from:
https://github.com/kikibeumer/Data-Quality-analysis-if-AIS-dataset/blob/main/AIS_Message_Decoder_Kiki_Beumer.ipynb
Python code used for this report on Github. Retrieved from:
https://github.com/kikibeumer/Data-Quality-analysis-if-AIS-dataset/tree/main
Danish Maritime Authority - AIS. Retrieved from:
https://github.com/dma-ais
Automatic identification system. Retrieved from:
https://en.wikipedia.org/wiki/Automatic_identification_system
AIS data Danish Maritime Authority. Retrieved from:
https://www.dma.dk/safety-at-sea/navigational-information/ais-data
GH AIS Message Format. Retrieved from:
https://www.iala-aism.org/wiki/iwrap/index.php/GH_AIS_Message_Format
AIVDM/AIVDO protocol decoding. Retrieved from:
https://gpsd.gitlab.io/gpsd/AIVDM.html
pyais. Retrieved from:
https://github.com/M0r13n/pyais/tree/master?tab=readme-ov-file
ais-protocol-decoding. Retrieved from:
https://github.com/doron2402/ais-protocol-decoding
AIRU Maidenhead Grid Locator Retrieved from:
https://www.mapability.com/ei8ic/maps/gridworld.php